Model and Algorithms for Point Cloud Construction Using Digital Projection Patterns

Tao Peng * Satyandra K. Gupta †

Abstract

This paper describes a computational framework for constructing point clouds using digital projection patterns. The basic principle behind the approach is to project known patterns on the object using a digital projector. A digital camera is then used to take images of the object with the known projection patterns imposed on it. Due to the presence of 3-D faces of the object, the projection patterns appear distorted in the images. The images are analyzed to construct the 3-D point cloud that is capable of introducing the observed distortions in the images. The approach described in this paper presents three advances over the previously developed approaches. First, it is capable of working with the projection patterns that have variable fringe widths and curved fringes and hence can provide improved accuracy. Second, our algorithm minimizes the number of images needed for creating the 3-D point cloud. Finally, we use a hybrid approach that uses a combination of reference plane images and estimated system parameters to construct the point cloud. This approach provides good run-time computational performance and simplifies the system calibration.

*Dept. of Mechanical Engineering, Univ. of Maryland, College Park, MD 20742. Email: pengtao@glue.umd.edu

†Dept. of Mechanical Engineering and Institute for Systems Research, University of Maryland, College Park, MD 20742. Email: skgupta@eng.umd.edu

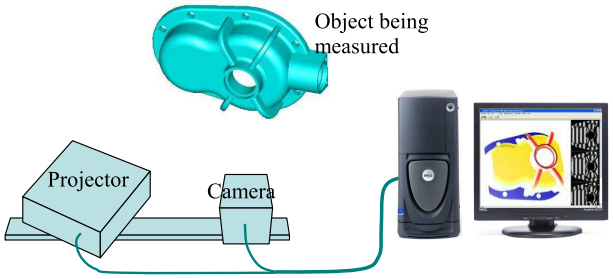
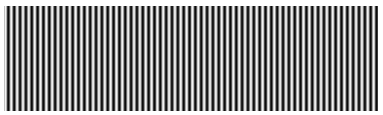


Figure 1: Schematic diagram of a PCCDFP system with one projector and one camera

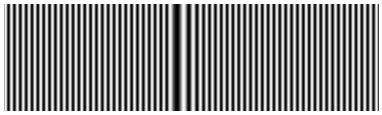
1 Introduction

Point cloud construction based on digital fringe projection (PCCDFP) is a technique for non-contact shape measurement. Due to its fast speed, flexibility, low cost and potentially high accuracy, PCCDFP has shown considerable promise in 3-D shape measurement, especially for applications that require acquisition of dense point clouds. The simplest PCCDFP system contains one projection unit and one camera (as shown in Fig. 1). The basic principle behind the approach is to project known patterns on the object using a digital projector. A digital camera is then used to take images of the object with the known projection patterns imposed on it. Due to the presence of 3-D faces of the object, the projection pattern appears distorted in the images. The images are analyzed to construct the 3-D point cloud that represents the shape of the object. Usually multiple projection patterns are used in a single measurement step to enhance accuracy.

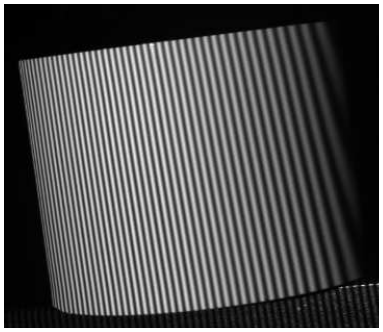
PCCDFP technique uses a set of projection patterns to obtain an absolute phase map of the object being measured. The measurement speed is determined by the hardware performance of the PCCDFP system as well as the number of projection patterns used in a measurement. Therefore, given similar measurement accuracy, algorithms using fewer projection patterns are favorable. A popularly used projection pattern for PCCDFP technique is sinusoidal fringe pattern (as shown in Fig. 2(a)), in which the fringe width is fixed. However, in certain situations, the fixed width fringe pattern does not give ideal results. Figure 2(c) shows the image of a cylinder under the projection of a sinusoidal fringe pattern. As can



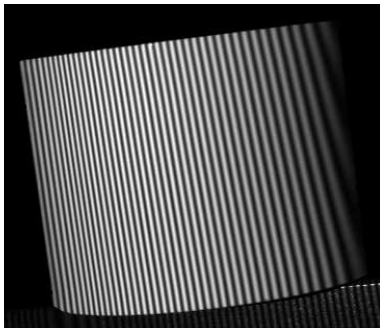
(a) Sinusoidal fringe pattern (fixed fringe width)



(b) Projection pattern with variable fringe width



(c) Image under the sinusoidal fringe pattern shown in (a)



(d) Image under the variable width fringe pattern shown in (b)

Figure 2: Measuring a cylinder using different fringe patterns: fixed fringe width vs. variable fringe width

be seen that, at the side of the cylinder fringes are crowded together and indistinguishable, which will cause significant error in the acquired phase map. On the other hand, with the use of variable fringe width in the pattern, the resulting image shows uniformly distributed fringes and the accuracy of measurement can be improved (see Fig. 2(b) and 2(d)). We should note that the problem with crowding of fringes in the image cannot be simply solved by using sinusoidal fringe pattern with a much larger fringe width since that will reduce the measurement accuracy in the middle portion of the object. Hence, in this paper we describe new model and algorithms for using variable width fringe patterns.

2 Related Work

The generation of point clouds using PCCDFP technique requires a mathematical model of the PCCDFP system, as well as associated algorithms for the computation of the 3-D coordinates of the points. Such mathematical models usually consist of a number of parameters which need to be acquired from calibration procedures. Two major types of mathematical

models exist for PCCDFP systems, namely optical geometry based and calibration matrix based [1].

The optical geometry based models describe the relationship between the image and the surface shape of the object using a number of equations and parameters that are derived from the optical geometry models of the camera and the projector. Early work in this field include the model proposed by Toyooka and Iwaasa [2] and the model by Hu et al. [3, 4]. Recently, Legarda-Sáenz et al. [5] adopted a full pinhole camera model with consideration of lens distortion in the modeling of PCCDFP systems. The model consists of 32 parameters in total, which allows maximal flexibility in system configuration and provides an accurate modeling of the system's optical geometry.

Calibration matrix based models use coefficients matrices to describe the measurement volume, which is defined as the intersection of the camera's field-of-view and the projection space of the projector. In the model proposed by Sitnik et al. [6], the coefficients matrices have the same dimensions as the images taken by the camera. Corresponding to each pixel in the image, there are three polynomial functions defined. One function is to convert phase-shift values to Z-coordinates of points in the point cloud. The other two functions are used to compute the X- and Y-coordinates from the Z-coordinates of points. The acquisition of the coefficients matrices is done by measuring a special calibration plate at a number of parallel positions (typically over 12). Compared to optical geometry based approach, calibration matrix based approach is simpler and easier to implement. However, it has two major disadvantages: 1) By representing the measurement volume using polynomial functions with limited orders, the model introduces truncation errors; 2) The calibration procedure needs to be completely redone each time the optical setup of a system changes, even if the changes take place locally.

Besides the mathematical model and the associated algorithms for point cloud generation and parameter estimation, another important part of the PCCDFP technique is the acquisition of absolute phase maps of objects. The accuracy of phase maps directly affects the

accuracy of the point clouds generated. Research in this area has been focused on the phase shifting and the phase unwrapping methods. A desired method should be able to obtain an accurate absolute phase map using as few projection patterns as possible.

Hibino et. al. [7] studied the design of phase shifting algorithms for the suppression of phase errors caused by the imperfect sinusoidal waveform and inaccurate phase-shifting of fringe patterns. They pointed out that a $(2j + 3)$-step phase shifting is required to eliminate the effects of harmonic components in the waveform up to the jth order. Surrel [8] studied the same problem but from a different perspective. The projection patterns generated by LCD/DMD based digital projectors are pixelated. For sinusoidal fringe patterns, this pixelation effect causes a zigzag sinusoidal waveform, which in turn causes errors in the phase map acquired from phase shifting. Coggrave and Huntley [9] studied this problem by conducting experiments and found out that, by purposely defocusing the projection in the measurement volume the phase errors caused by the pixelation of fringe patterns can be reduced significantly.

Phase maps obtained from the phase shifting technique are "wrapped" phase maps, i.e. the phase values are wrapped in a 2π range and the fringe order information is lost. Phase unwrapping algorithms are used to recover the fringe order information and convert wrapped phase maps to absolute phase maps. Saldner and Huntley [10] proposed a temporal phase unwrapping algorithm which is able to deal with complicated phase jumps caused by surface discontinuities or shadows on surface. Zhao et al. [11] improved the temporal phase unwrapping approach by using fewer fringe patterns to achieve similar performance. An extension of Zhao's algorithm by using three different fringe frequencies is able to achieve both high accuracy and robustness in the acquisition of absolute phase maps [12, 5].

3 Overview of Approach

Our mathematical model for PCCDFP systems uses a full pinhole camera model (with consideration of lens distortions) for both the camera and the projector [5]. Using this model, we developed the following algorithms for shape measurement and system calibration:

- **Algorithm for construction of absolute phase map**: The phase map construction algorithm generates a set of fringe patterns for projection and constructs an absolute phase map of the object surface from the corresponding images that have been taken. The algorithm we developed uses multiple fringe frequencies but requires fewer projection patterns than conventional methods. It is defined on a generalized concept of phase shifting and hence is able to deal with fringe patterns with curved fringes and continuously varying fringe width.

- **Algorithm for construction of point cloud**: The point cloud construction algorithm converts the absolute phase map of the object into a dense 3-D point cloud that represents the object's surface. Although the mathematical model contains a large number of parameters, the point cloud construction algorithm does not require all of them. For the projector parameters, only the position of the projection center is needed. An absolute phase map of a (flat) reference plane, as well as the position and orientation of the plane, are needed in addition.

- **Algorithm for estimation of parameters**: The functionality of the parameter estimation algorithm is to accurately evaluate the parameters in the mathematical model, since most of them are related to optical characteristics and hence cannot be measured directly. The algorithm consists of two parts, camera calibration and the estimation of the position of the projector's projection center.

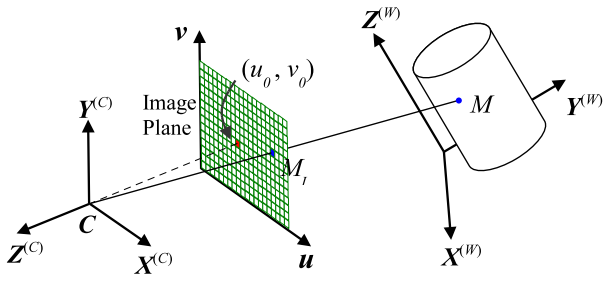


Figure 3: Pinhole camera model

4 Mathematical Model

4.1 Pinhole Camera Model

The pinhole camera model has been shown to be an accurate model for cameras and is widely used in areas such as photogrammetry and machine vision. A complete pinhole model with consideration of lens distortions can be described by a series of transformations presented below (see Fig. 3):

1. **Transformation from world coordinates to camera coordinates**: Camera coordinate frame (Cartesian) is defined by a camera's optics and image formation hardware (e.g. the CCD). The transformation of a 3-D point M's coordinates from the world coordinate frame to the camera coordinate frame can be described by the following equation:

$$\left[x_M^{(C)}, y_M^{(C)}, z_M^{(C)}\right]^T = \boldsymbol{R}_{W2C} \cdot \left[x_M^{(W)}, y_M^{(W)}, z_M^{(W)}\right]^T + \boldsymbol{T}_{W2C} \tag{1}$$

   where $\boldsymbol{R}_{W2C}$ is the rotation matrix and $\boldsymbol{T}_{W2C}$ is the translation vector.

2. **Perspective projection to the image plane**: As a convention, the camera's image plane is defined to be perpendicular to $Z^{(C)}$-axis and intersecting it at $z^{(C)} = -1$. The two axes of the image plane, $\boldsymbol{u}$ and $\boldsymbol{v}$, are parallel to $X^{(C)}$- and $Y^{(C)}$-axis respectively. Let M_I denote the projection of point M on the image plane. The coordinates of M_I

in the camera coordinate frame, i.e. $\left[x_M^{(n)}, y_M^{(n)}, -1\right]^T$, can be calculated as follows:

$$\left[x_M^{(n)}, y_M^{(n)}\right]^T = \left[\frac{x_M^{(C)}}{-z_M^{(C)}}, \frac{y_M^{(C)}}{-z_M^{(C)}}\right]^T \tag{2}$$

3. **Lens distortion model**: Due to distortions of optical lenses, the true position of M_I is usually off its nominal position $\left[x_M^{(n)}, y_M^{(n)}, -1\right]^T$. According to the "Plumb Bob" model proposed by Brown [13], the true position of M_I, $\left[x_M^{(d)}, y_M^{(d)}, -1\right]^T$, can be calculated from the nominal position as follows:

$$\begin{bmatrix} x_M^{(d)} \\ y_M^{(d)} \end{bmatrix} = \left(1 + k_1 r^2 + k_2 r^4 + k_5 r^6\right) \begin{bmatrix} x_M^{(n)} \\ y_M^{(n)} \end{bmatrix} + \begin{bmatrix} 2k_3 x_M^{(n)} y_M^{(n)} + k_4 \left(r^2 + 2\left(x_M^{(n)}\right)^2\right) \\ k_3 \left(r^2 + 2\left(y_M^{(n)}\right)^2\right) + 2k_4 x_M^{(n)} y_M^{(n)} \end{bmatrix} \tag{3}$$

where $r^2 = \left(x_M^{(n)}\right)^2 + \left(y_M^{(n)}\right)^2$ and k_s $(s = 1, \ldots, 5)$ are the radial and tangential distortion coefficients of the lenses.

4. **Transformation from camera coordinates to pixel coordinates**: For digital cameras, the position of M_I is spatially digitized and presented in pixel coordinates. The homogeneous pixel coordinates of M_I, denoted by $\left[x_M^{(p)}, y_M^{(p)}, 1\right]^T$, can be calculated as follows:

$$\begin{bmatrix} x_M^{(p)} \\ y_M^{(p)} \\ 1 \end{bmatrix} = \boldsymbol{A} \cdot \begin{bmatrix} x_M^{(d)} \\ y_M^{(d)} \\ 1 \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} f_x & \alpha f_x & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4}$$

where $\boldsymbol{A}$ is the camera's intrinsic matrix, in which (u_0, v_0) is the pixel coordinates of the principal point, f_x and f_y are the scale factors for axes $\boldsymbol{u}$ and $\boldsymbol{v}$ respectively, and α is a coefficient describing the skewness of the axes $\boldsymbol{u}$ and $\boldsymbol{v}$.

5. **Inverse transformations (from pixel coordinates to world coordinates)**: Shape reconstruction is a process of computing the world coordinates of points from image(s).

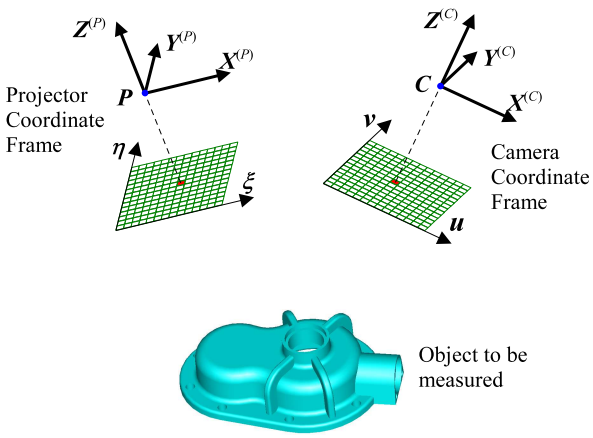


Figure 4: Mathematical model for PCCDFP system

Therefore, the inverse operations of the above procedures are required. We construct this model by computing inverse of the above four transformations.

4.2 Mathematical Model for the System

The mathematical model we used for the system consists of two parts, a camera model and a projector model. The camera model is a complete pinhole model with consideration of lens distortion, as explained above. Since a computer projector (LCD or DMD based) acts as an inverted digital camera from the optical geometry perspective, it can also be modeled accurately by a pinhole model as the camera does. In the system model (see Fig. 4), a Cartesian coordinate frame $X^{(P)}Y^{(P)}Z^{(P)}$ is defined for the projector. The image plane defined by $\boldsymbol{\xi}$–$\boldsymbol{\eta}$ axes represents the LCD/DMD chip of the projector. The pixel coordinates of the principal point in the image plane is (ξ_0, η_0). All transformations related to the projector model are analogous to the ones defined for the camera (see Section 4.1).

5 Algorithm for Construction of Absolute Phase Map

5.1 Phase Shifting Using Generalized Fringe Patterns

The concept of generalized fringe pattern is introduced to distinguish itself from sinusoidal fringe pattern. A sinusoidal fringe pattern contains straight fringes and has a sinusoidal intensity profile at cross-sections, as the one shown in Fig. 2(a). A generalized fringe pattern is a fringe pattern in which the fringes may be curved and the fringe width may vary continuously throughout the pattern. The projection patterns shown in Fig. 2(b), 11(b) and 10(b) are all examples of generalized fringe patterns.

The algorithm we developed for absolute phase map construction uses generalized fringe patterns. The conventional phase shifting technique is revised to incorporate the use of generalized fringe patterns. By using a new mathematical description, generation of a set of fringe patterns for absolute phase measurement using phase shifting technique can be presented as a two-step process as follows:

- **Step 1**: A phase function $\Phi(\xi, \eta)$ is defined in the image plane of the projector, i.e. the $\boldsymbol{\xi}$–$\boldsymbol{\eta}$ plane. $\Phi(\xi, \eta)$ must be continuous in the $\boldsymbol{\xi}$–$\boldsymbol{\eta}$ plane and monotonic in either $\boldsymbol{\xi}$ or $\boldsymbol{\eta}$ direction, depending on the position of the camera w.r.t. the projector. These two constraints are required by the point cloud construction algorithm, which will be explained later.

- **Step 2**: The phase function $\Phi(\xi, \eta)$ is converted to light projection pattern(s) by some encoding method. Light properties that can be used for encoding include intensity, color, etc. In practice, light intensity, without the involvement of color, is most popularly used since the intensity of light can be measured accurately by photo sensors such as CCD. A widely used modulation function for converting phase values to light intensities is the sinusoidal function. For digital projectors, in which light intensity is presented in gray-levels, the sinusoidal modulation can be described using the following

equation:

$$I^{(P)}(\xi,\eta) = \frac{I^{(P)}_{max}}{2}\left[1+\sin(\Phi(\xi,\eta))\right] \tag{5}$$

where $I^{(P)}(\xi,\eta)$ is the gray-level in the projection pattern and $I^{(P)}_{max}$ is the maximum gray-level of the projector.

By using the sinusoidal modulation, a large variety of fringe patterns can be defined by appropriately constructed phase functions. For example, a vertical sinusoidal fringe pattern (as the one shown in Fig. 2(a)) can be defined by the following phase function

$$\Phi(\xi,\eta) = c \cdot \xi \tag{6}$$

where c is a constant.

5.2 Construction of Absolute Phase Map

The functionality of the phase map construction algorithm is to generate an absolute phase map from a set of images obtained from phase shifting. As mentioned in the Related Work, the phase map construction algorithm with the use of multiple fringe frequencies has been used in many latest research work on PCCDFP technique [5, 12]. For brevity, we will refer to this algorithm as MFF in the following context. This algorithm provides high accuracy in phase measurement, is robust to noises (in projections and images) and can handle all sorts of surface discontinuities. Here we propose a modified version of this approach, which achieves similar performance but uses fewer projection patterns. By using extended mathematical definitions, our algorithm is also able to deal with generalized fringe patterns.

A common scheme of the conventional MFF algorithm uses 4-step phase shifting and 3 fringe frequencies, meaning a total number of 12 phase-shifted fringe patterns. Our algorithm

uses 8 projection patterns instead, which can be defined using the following equation:

$$I_n^{(P),k}(\xi,\eta) = \frac{I_{max}^{(P)}}{2}\left[1+\sin\left(c_k\cdot\Phi^{(P)}(\xi,\eta)+\frac{(n-1)\pi}{2}\right)\right] \tag{7}$$

where $k=0,1,2$ represent different levels of fringe patterns, $n=1,\ldots,4$ represent different phase-shift values, (ξ,η) is the pixel coordinates in the projection pattern, $\Phi^{(P)}(\xi,\eta)$ is the phase function, c_k is a scale coefficient, $I_{max}^{(P)}$ is the maximum intensity in the patterns, and $I_n^{(P),k}(\xi,\eta)$ is the intensity value of pixel (ξ,η).

For any phase function $\Phi^{(P)}(\xi,\eta)$, Eqn. 7 defines a set of 8 fringe patterns. An example of such fringe pattern set is shown in Fig. 5. The 8 patterns are divided into 3 levels by phase sensitivity (corresponding to the scale coefficient c_k). Level-0 has the lowest phase sensitivity (i.e. c_0 is the smallest in $\{c_k\}$, $k=0,1,2$) and contains two patterns with 0 and $\pi/2$ phase-shift respectively (corresponding to $n=1,2$). Level-2 has the highest phase sensitivity and contains four patterns corresponding to $n=1$ to 4. Level-1 has the intermediate phase sensitivity and also contains two patterns ($n=1,2$). The concept of different "level" of fringe patterns is an analogy to the "fringe frequency" in the MFF algorithm. We introduced the "level" concept to avoid confusion with "fringe frequency", since in a generalized fringe pattern the fringe width, and hence "fringe frequency", may vary continuously throughout the pattern.

Projecting the 8 fringe patterns as defined above onto the object surface sequentially, the corresponding images of the object can be described using the following equation:

$$I_n^{(k)}(i,j) = A(i,j)+B(i,j)\sin\left(\Phi^{(k)}(i,j)+\frac{(n-1)\pi}{2}\right) \tag{8}$$

where (i,j) is the pixel indices in the image, $k=0,1,2$ represent different levels of fringe patterns and $n=1,\ldots,4$ represent different phase-shift values. For each pixel (i,j), $I_n^{(k)}(i,j)$ is its gray-scale intensity in the image that corresponds to the projection pattern at level k

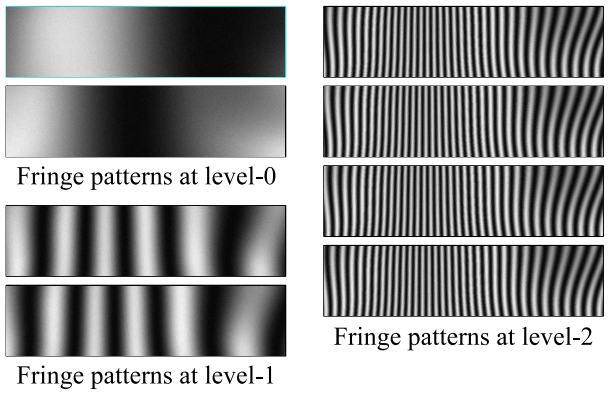


Figure 5: A set of 8 (generalized) fringe patterns used in phase map construction

and n-th phase-shift; $\Phi^{(k)}(i,j)$ is the absolute phase value of the pixel's corresponding point on the object surface. Assuming that both the camera and the projector have a fairly large depth-of-view and the reflection of the object surface is linear, $A(i,j)$ and $B(i,j)$ are both constants for pixel (i,j) in all 8 images.

The procedure to compute the absolute phase map of the object surface by using the 8 images is described below, in which we use Φ for absolute phase maps and ϕ for wrapped phase maps.

Firstly, the wrapped phase map $\phi^{(2)}$ is computed using the following equation:

$$\phi^{(2)}(i,j) = \arctan\left(\frac{I_1^{(2)}(i,j) - I_3^{(2)}(i,j)}{I_2^{(2)}(i,j) - I_4^{(2)}(i,j)}\right) \tag{9}$$

The coefficients $A(i,j)$ can be calculated by

$$A(i,j) = \frac{1}{4}\left[I_1^{(2)}(i,j) + I_2^{(2)}(i,j) + I_3^{(2)}(i,j) + I_4^{(2)}(i,j)\right] \tag{10}$$

Using the $A(i,j)$ calculated, phase map $\phi^{(0)}$ and $\phi^{(1)}$ can be computed from

$$\phi^{(k)}(i,j) = \arctan\left(\frac{I_1^{(k)}(i,j) - A(i,j)}{I_2^{(k)}(i,j) - A(i,j)}\right), \quad k = 0, 1 \tag{11}$$

Notice that $\phi^{(0)}$ is an absolute phase map because it is obtained from the level-0 fringe patterns, which are designed to have a low phase sensitivity, e.g. phase range within $[-\pi, \pi]$, to avoid phase wrapping. To make the notations consistent, we depict the fact as $\Phi^{(0)} = \phi^{(0)}$. From the relationship between the fringe patterns at different levels (see Eqn. 7), we can get that, for any pixel (i, j)

$$\begin{aligned} \Phi^{(1)}(i,j) &= (c_1/c_0) \cdot \Phi^{(0)}(i,j) \ , \\ \Phi^{(2)}(i,j) &= (c_2/c_1) \cdot \Phi^{(1)}(i,j) \end{aligned} \tag{12}$$

where c_0, c_1 and c_2 are coefficients defined in Eqn. 7. It can also be known from the relationship between an absolute phase map and its wrapped counterpart that

$$\Phi^{(k)}(i,j) = \phi^{(k)}(i,j) + 2\pi \cdot n_k(i,j) \ , \quad k = 1, 2 \tag{13}$$

where $n_k(i, j)$ are unknown integers. By combining Eqn. 12 and 13, the absolute phase map $\Phi^{(1)}(i, j)$ and $\Phi^{(2)}(i, j)$ can be solved one after another. $\Phi^{(2)}(i, j)$ has the highest phase sensitivity and will be provided to the point cloud construction algorithm as input.

The idea to reduce number of projection patterns required in phase map construction by utilizing $A(i, j)$ can be applied to any phase shifting strategy. For the generic N-step phase shifting, whose images can be described by the following equation

$$I_n(i,j) = A(i,j) + B(i,j) \sin\left(\Phi(i,j) + 2\pi \frac{n-1}{N}\right) \ , \ \ n = 1, \ldots, N \tag{14}$$

the corresponding equation for computing $A(i, j)$ is as follows:

$$A(i,j) = \frac{1}{N} \sum_{n=1}^{N} I_n(i,j) \tag{15}$$

Notice that although assumed to be a constant for all 8 images, $B(i, j)$ (as defined in Eqn. 8) is not utilized to further reduce the number of patterns required in phase map

construction. This is because, in practice, the value of $B(i, j)$ may vary slightly for different level of fringe patterns due to the out-of-focus projection on the object surface. However, this does not affect the value of $A(i, j)$ at all.

6 Algorithm for Construction of Point Cloud

The point cloud construction algorithm converts the absolute phase map of an object to a dense 3-D point cloud that represents the object's surface. The algorithm requires the following data to be known a priori:

- All intrinsic parameters of the camera, i.e. the intrinsic matrix $\boldsymbol{A}$ and the lens distortion coefficients k_s $(s = 1, \ldots, 5)$. Definitions of the parameters can be found in Section 4.1.

- The position of the projector's projection center, $P(P_x, P_y, P_z)$, with respect to the camera coordinate frame. (All spatial coordinates referred in this section are w.r.t. the camera's coordinate frame if not otherwise declared)

- The absolute phase map of a reference plane, denoted by $\Phi_R(u, v)$, and the position and orientation of the reference plane, which is represented by a point on the plane, O_R, and the plane's normal vector $\boldsymbol{n}_R$.

Let $\Phi(u, v)$ denote the absolute phase map of the object being measured. For each pixel (u, v) with a valid absolute phase value, a 3-D point can be possibly generated. The algorithm for computing the 3-D coordinates of point M, corresponding to pixel (u, v), is described at below. Figure 6 shows an illustration of the geometric relationships involved in the computation.

Let point M_I denote the idealized position of the center of pixel (u, v) on the image plane of the camera. It can be known from the pinhole camera model that point M lies on line

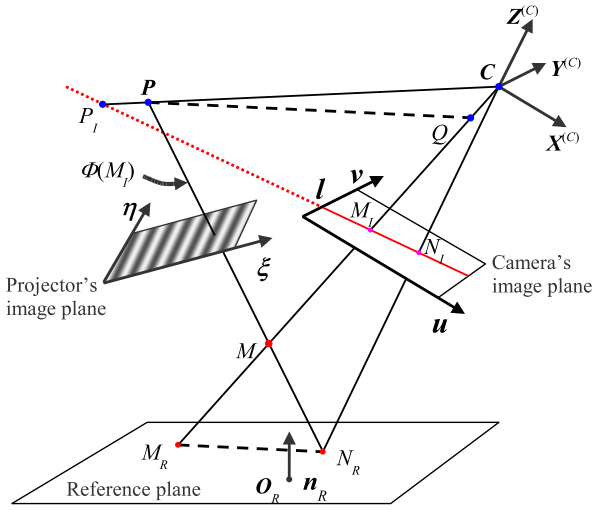


Figure 6: Construction of point cloud

$\overline{CM_I}$. Let point M_R denote the intersection of line $\overline{CM_I}$ with the reference plane. Similarly, let point N_R denote the intersection of line $\overline{PM}$ with the reference plane and let N_I denote the intersection of line $\overline{CN_R}$ with the image plane of the camera. Define point Q as the intersection of line $\overline{CM_I}$ with the plane that is parallel to the reference plane while passes through point P. It can be seen from the definitions that, points C, P, Q, M_I, N_I, M_R, N_R and M are coplanar.

The procedure to compute the 3-D position of point M is as follows:

1. As a first step, locate the pixel coordinates of point N_I by searching in the reference phase map $\Phi_R(u, v)$ along line $\overline{P_I M_I}$ for a point with phase value $\Phi(M)$ (see Fig. 7), in which P_I is the projection of point P on the camera's image plane $\boldsymbol{u}$–$\boldsymbol{v}$. Due to the restrictions imposed on the projection patterns used in phase-shifting (see Section 5.1), the distribution of phase values along line $\overline{P_I M_I}$ is guaranteed to be monotonic. Hence, the location of point N_I is unique if one exists.

2. The 3-D coordinates of points M_I and N_I are calculated from their pixel coordinates, by using the pinhole camera model and the camera's intrinsic parameters (see Section 4.1). Accordingly, the 3-D coordinates of points M_R and N_R can be calculated from line-

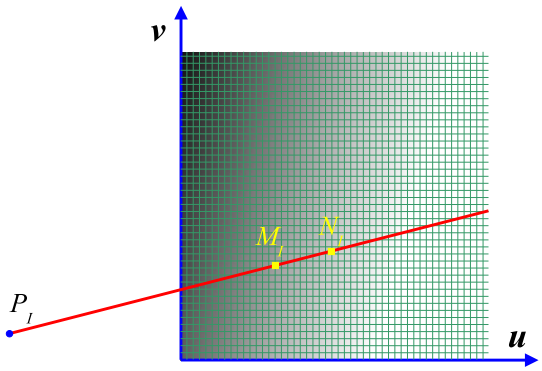


Figure 7: Find the pixel coordinates of N_I in the reference phase map $\Phi_R(u, v)$

plane-intersections by using the following equations:

$$M_R = C + \left(\frac{\overrightarrow{CO_R} \cdot \boldsymbol{n}_R}{\overrightarrow{CM_I} \cdot \boldsymbol{n}_R} \right) \overrightarrow{CM_I} \,, \quad N_R = C + \left(\frac{\overrightarrow{CO_R} \cdot \boldsymbol{n}_R}{\overrightarrow{CN_I} \cdot \boldsymbol{n}_R} \right) \overrightarrow{CN_I} \tag{16}$$

where O_R and $\boldsymbol{n}_R$ represent the position and orientation of the reference plane. Similarly, the 3-D coordinates of point Q can be calculated from the following equation:

$$Q = C + \left(\frac{\overrightarrow{CP} \cdot \boldsymbol{n}_R}{\overrightarrow{CM_I} \cdot \boldsymbol{n}_R} \right) \overrightarrow{CM_I} \tag{17}$$

3. Finally, the 3-D position of point M can be calculated from the triangular similarities between $\triangle M M_R N_R$ and $\triangle MQP$ as follows:

$$M = Q + \left(\frac{\left|\overrightarrow{PQ}\right| \left|\overrightarrow{QM_R}\right|}{\left|\overrightarrow{PQ} + \overrightarrow{M_R N_R}\right| \left|\overrightarrow{CM_I}\right|} \right) \overrightarrow{CM_I} \tag{18}$$

The point cloud construction algorithm proposed here is a hybrid of the optical geometry based approach and the calibration matrix based approach. It requires full knowledge of the camera parameters, as well as the geometric information of the reference plane. For the projector parameters, only the position of the projection center, P, is needed explicitly. The rest of the geometric information about the projector is given by the reference phase

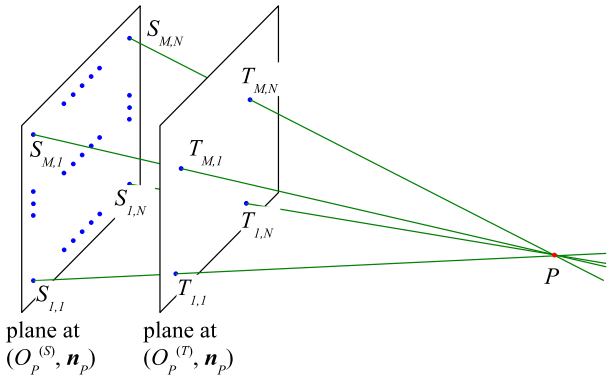
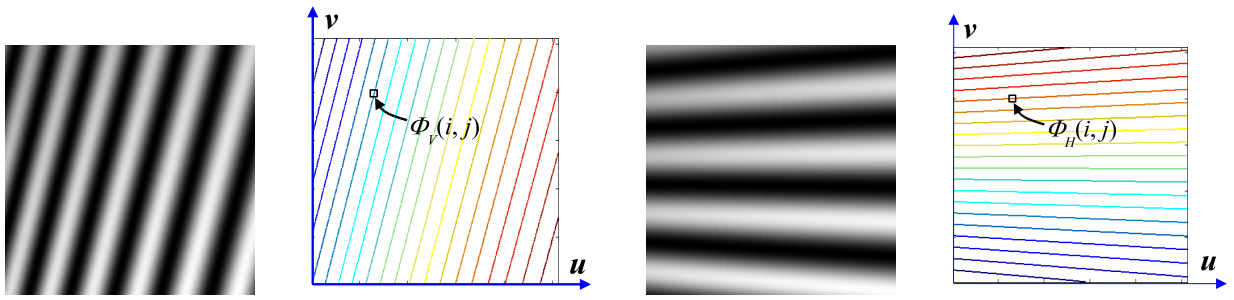


Figure 8: Estimation of the projector's projection center

map in an implicit manner. Compared to the pure optical geometry based approach, this hybrid approach requires fewer parameters (9 parameters less compared to Legarda-Sáenz's method [5]). As a result, the parameter estimation algorithm becomes simpler and numerically more stable, since there are fewer variables in the nonlinear minimization process. Although a reference phase map is required in addition, it can be easily acquired by projecting sinusoidal fringe patterns to a flat plane. The hybrid approach does not lose any generality or accuracy by ignoring the majority of projector parameters, as all system parameters are reflected in the algorithm, explicitly or implicitly. Compared to the calibration matrix based approach, this hybrid approach is more accurate since it utilizes an accurate camera model instead of approximate polynomial interpolations for the construction of the point cloud. Moreover, it requires much less memory (and storage space) for the computation and the calibration process is made easier and more flexible.

7 Algorithm for Estimation of System Parameters

The parameter estimation algorithm can be divided into two parts, the estimation of camera parameters and the estimation of the position of the projector's projection center. We used a camera calibration toolbox developed by the computer vision group at the California Institute of Technology [14] for the acquisition of camera parameters.



(a) One of the images for acquiring Φ_V

(b) Contours of the phase map Φ_V

(c) One of the images for acquiring Φ_H

(d) Contours of the phase map Φ_H

Figure 9: Acquisition of the absolute phase maps $\Phi_V(u, v)$ and $\Phi_H(u, v)$ by using vertical and horizontal fringe patterns respectively

As mentioned earlier, a projector is very similar to a camera in terms of optical geometry and can be described accurately by a pinhole camera model. The parameter estimation algorithm for projector can hence be developed by adapting the existing camera calibration methods. However, such algorithms have rarely been discussed in the research of PCCDFP technique until recently in a paper by Legarda-Sáenz et al. [5]. We propose an algorithm for the estimation of the position of the projection center (point $P(P_x, P_y, P_z)$ as shown in Fig. 6), since it is the only projector parameter that is required by the point cloud construction algorithm. This algorithm requires all camera parameters to be known, therefore a camera calibration needs to be performed beforehand. The calibration procedure, which involves measuring a flat plane at two or more parallel positions, is described in details below.

- At the first step, a flat calibration plate is placed at the farther side of the measurement volume. The position and orientation of the plate, denoted as $(O_P^{(S)}, \boldsymbol{n}_P)$, can be estimated by using the calibrated camera. Two absolute phase maps of the calibration plate, $\Phi_V^{(S)}(u, v)$ and $\Phi_H^{(S)}(u, v)$, are obtained by projecting a set of patterns with vertical and horizontal fringes respectively (as shown in Fig. 9). For each pixel (i, j) in the camera's image plane, its corresponding point on the plane $(O_P^{(S)}, \boldsymbol{n}_P)$, denoted as $S_{i,j}$, can be located by using the camera model and parameters (see Fig. 8). By the acquisition of phase map $\Phi_V^{(S)}$ and $\Phi_H^{(S)}$, each point $S_{i,j}$ is associated with two phase

values, $\Phi_V^{(S)}(i,j)$ and $\Phi_H^{(S)}(i,j)$. The phase value pair $(\Phi_V^{(S)}, \Phi_H^{(S)})$ determines a single point on the image plane of the projector.

- At the second step, the calibration plate is translated to $(O_P^{(T)}, \boldsymbol{n}_P)$, which is at the nearer side of the measurement volume. The precise values of $O_P^{(T)}$ can be either obtained directly from the camera or calculated from the translation which can be measured by other means. Again, two absolute phase maps of the calibration plate are acquired, namely $\Phi_V^{(T)}(u,v)$ and $\Phi_H^{(T)}(u,v)$.

- For each point $S_{i,j}$ on plane $(O_P^{(S)}, \boldsymbol{n}_P)$, whose phase value pair is $(\Phi_V^{(S)}, \Phi_H^{(S)})$, we can locate a point on plane $(O_P^{(T)}, \boldsymbol{n}_P)$, who has the same phase value pair as $S_{i,j}$ does. Denote this point as $T_{i,j}$ and its projection on the camera's image plane as $T_{i,j}^{(I)}$. Notice that the subscripts (i,j) of $T_{i,j}$ is for reference to point $S_{i,j}$, not the corresponding pixel coordinates of T. The location of $T_{i,j}^{(I)}$ in the image plane $\boldsymbol{u}$–$\boldsymbol{v}$ is done by finding the intersection of the two contour lines, $\Phi_V = \Phi_V(S_{i,j})$ and $\Phi_H = \Phi_H(S_{i,j})$. Once $T_{i,j}^{(I)}$ is located, the position of $T_{i,j}$ can be calculated from the camera model and parameters.

- Since point $S_{i,j}$ and point $T_{i,j}$ share the same phase value pair, line $\overline{S_{i,j}T_{i,j}}$ must pass through the projection center P, according to the pinhole camera model. Therefore, the position of projection center P can be located by the intersection of all such lines that connect a point on plane $(O_P^{(S)}, \boldsymbol{n}_P)$ with its corresponding point on plane $(O_P^{(T)}, \boldsymbol{n}_P)$. The points on plane $(O_P^{(S)}, \boldsymbol{n}_P)$ that do not have corresponding points due to the shifted phase range at different plane positions, are skipped. Since the intersection of multiple lines is an over-constrained problem, the position of point P is computed in a least-square manner.

8 Implementation and Results

Based on the proposed mathematical model and algorithms, a software system was developed for shape reconstruction. A PCCDFP hardware was also built, which consists of a commercial computer projector (BenQ PB2220, DMD based, 1024×768 resolution, 1700 ANSI lumens) and a B/W digital camera (1/3" Sony CCD, 640×480 resolution). Using the developed system, a number of shape measurement experiments were performed on various objects. We will describe few representative results.

Fig. 10 shows CAD model of a spline surface. Shape measurements were performed on this spline surface by generating simulated images under structured lighting. We have also developed an algorithm for automatically generating adaptive projection patterns for objects of different shapes to optimize measurement performance. A detailed description of this algorithm will be presented in a future paper [15]. Fig. 10(c) shows the measurement performance results, in terms of the size of unresolvable area and the RMS value of measurement errors. For fixed-pitch fringe patterns (shown in blue crosses), a high fringe number (n_F, which corresponds to a smaller fringe pitch) can achieve good measurement accuracy but sacrifices measurement coverage; while a low fringe number has excellent measurement coverage but the accuracy is poor. With the use of adaptive projection pattern (shown in red triangles), the overall performance both in terms of accuracy and measurement coverage is improved.

Fig. 11 shows measurement results on a physical flower pot. The height of this flower pot was approximately $155mm$. As shown in this figure, the performance in terms of measurement coverage again improves as a result of using adaptive patterns. However, in this case the comparison of measurement accuracy is not available since we do not have an accurately measured model of the flower pot.

The system worked very well in dealing with objects with significant surface discontinuities. For example, the measurements of a plastic drill housing and the base of a telephone,

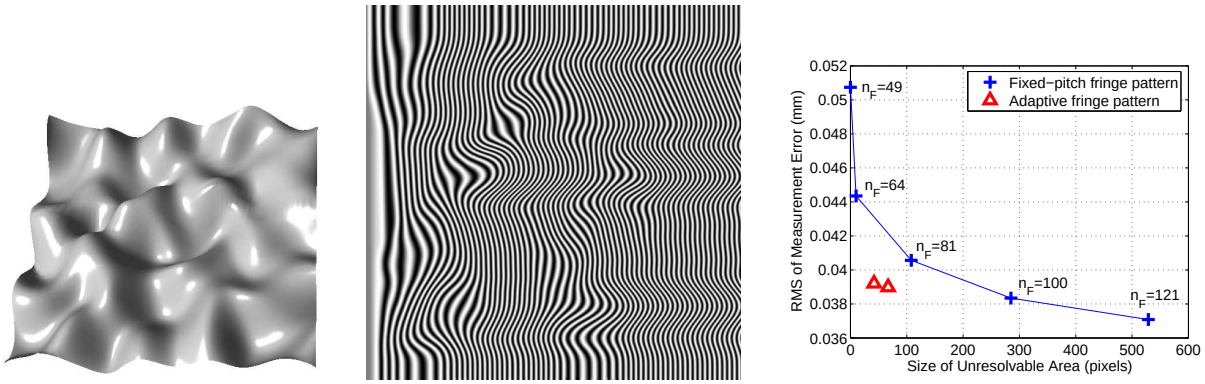
are presented in Fig. 12 and 13. Our system is capable of correctly constructing the model of the all the visible surfaces on these complex objects. We have achieved similar results on two dozen other parts of comparable complexity. For the sake of brevity those results will not be reported in this paper.

To formally assess the measurement accuracy of our system, two gauge parts with accurately known geometry were measured (see Fig. 14 and 15). Before performing the measurements, a paint coating was applied to these metal parts to avoid specular reflections. A photograph of the first gauge part after painting is shown in Fig. 14(a). The point cloud acquired from the measurement was aligned to the CAD model of the part (see Fig. 14(b)) and the divergence between the two was analyzed (see Fig. 14(c)). The result shows that the developed system reaches an average accuracy of $75\mu m$ over a measurement volume of $280mm \times 220mm \times 250mm$, which we believe can be further improved by a more accurate system calibration.

Currently, the average time to acquire images and construct point clouds is 5 seconds. In these experiments, the computation was performed on a Pentium M (1.6GHz) computer with 512MB memory. By using the new phase map construction algorithm, the time required for one measurement is reduced by 33% comparing to the traditional method, since fewer projection patterns are needed.

9 Conclusions

This paper describes a computational framework for constructing point clouds using digital projection patterns. The approach described in this paper presents three advances over the previously developed approaches. First, it is capable of working with generalized fringe patterns that have variable fringe widths and hence can provide improved accuracy. Second, by fully exploiting the information in the images obtained from phase-shifting, our algorithm for phase map construction uses fewer projection patterns than existing algorithms without



(a) Picture of the spline surface

(b) Automatically generated adaptive projection pattern

(c) Measure performance: adaptive pattern vs. fixed-pitch fringe patterns

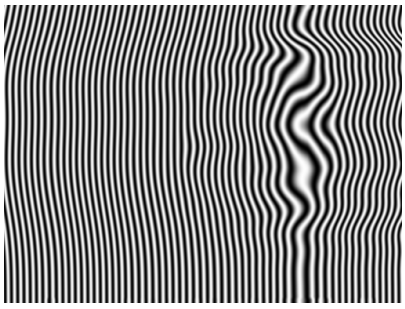
Figure 10: Measurement of a spline surface

compromising performance. Finally, the algorithm for point cloud construction requires fewer sensor parameters than similar algorithms in its class, e.g. algorithm proposed by Legarda-Sáenz et. al. [5]. Hence, it simplifies system calibration without compromising accuracy. The new algorithm also handles projector's lens distortions automatically. Our tests also show that the approach described in this paper provides good run-time computational performance and is capable of handling objects with complex surfaces that involve discontinuities and shape corners.

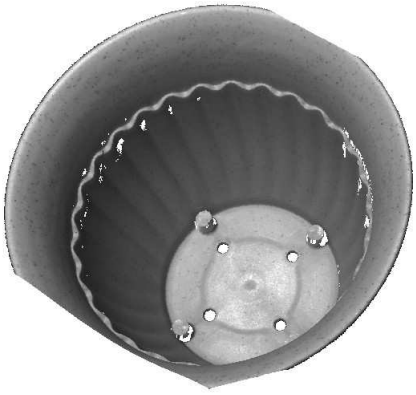
The approach described in the paper generates accurate and dense point clouds on a regular grid. Potential applications of the algorithms described in this paper include the following. First, the point clouds can be used for performing inspection and hence performing quality control. Second, the point clouds can be used to perform surface reconstruction for virtual and physical prototyping. Third, the point clouds can be used for designing molds and patterns for replicating artwork. Finally, the point clouds can be used for machining a replacement part for a broken or damaged part for which a CAD model does not exist.

In future, we plan to characterize the effects of system and operational parameters on the achieved accuracy. This characterization will allow us to select the patterns to optimize the accuracy. In addition, this information will be used for designing system to optimize its

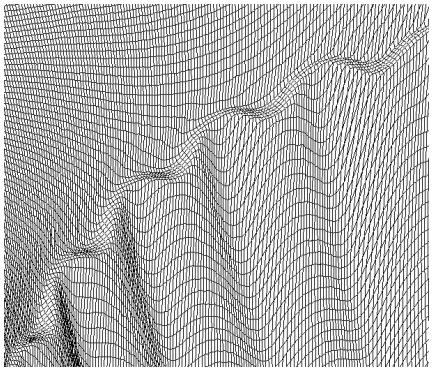
(a) Photograph of the flower pot



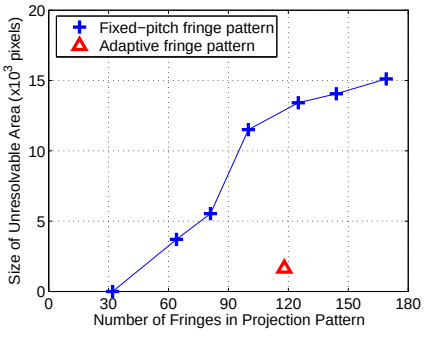
(b) Automatically generated adaptive projection pattern



(c) Rendering of the point cloud acquired



(d) Enlarged view of a mesh rendering of the point cloud



(e) Measurement coverage: adaptive pattern vs. fixed-pitch fringe patterns

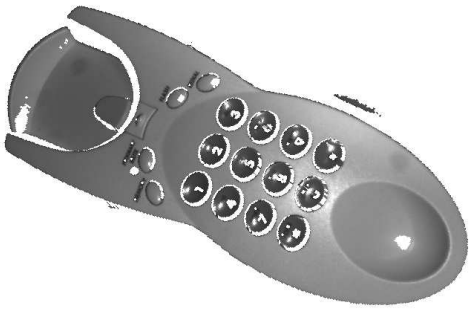
Figure 11: Measurement of a plastic flower pot

(a) Photograph of the drill housing

(b) Rendering of the point cloud acquired

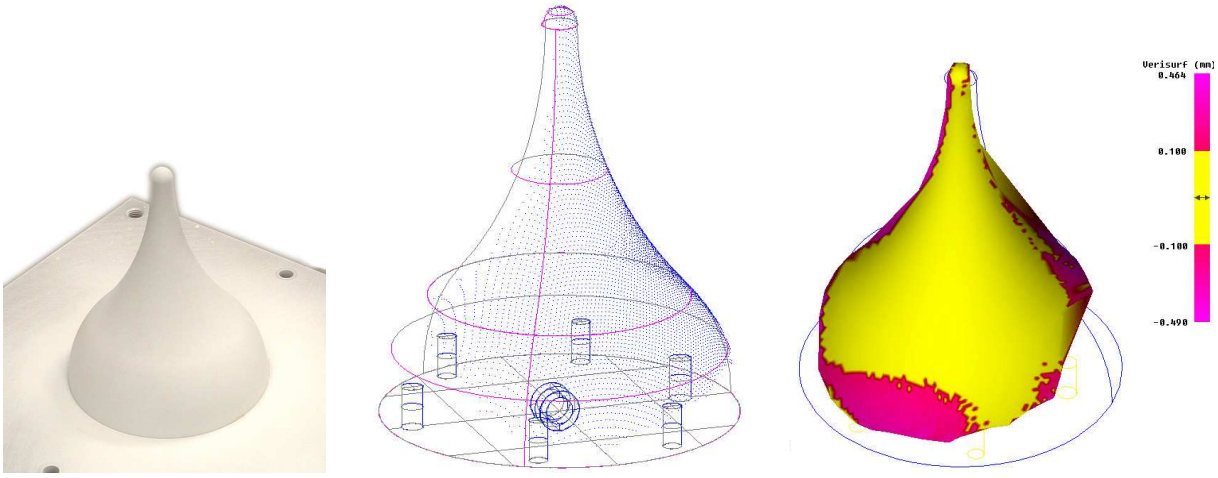
Figure 12: Measurement of a plastic drill housing

(a) Photograph of the telephone



(b) Rendering of the point cloud acquired

Figure 13: Measurement of the base of a telephone

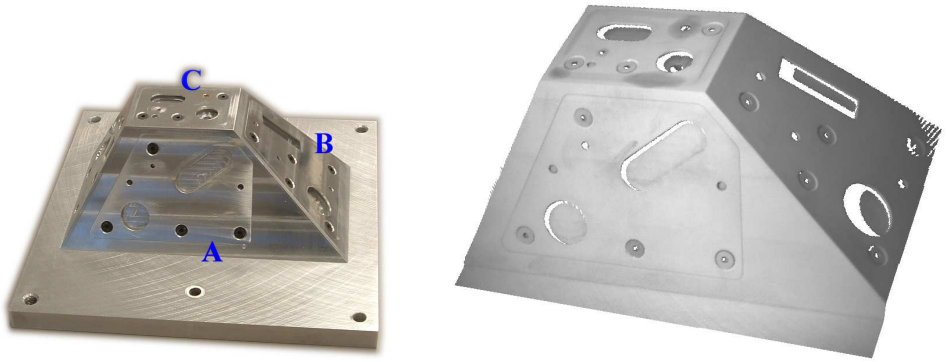


(a) Photograph of the gauge part

(b) The acquired point cloud after being aligned with the CAD model of the part

(c) Distribution of the divergence between the point cloud and the CAD model

Figure 14: Measurement of gauge part #1



(a) Photograph of the gauge part

(b) Rendering of the point cloud acquired

Figure 15: Measurement of gauge part #2

versatility and performance.

10 Acknowledgments

This work has been supported by NSF grant DMI-0093142. However, the opinions expressed here are those of the authors and do not necessarily reflect that of the sponsor.

References

[1] B. Breuckmann, F. Halbauer, E. Klaas, and M. Kube. 3D-Metrologies for industrial applications. In *Proceedings of SPIE 3102*, pages 20–29, 1997.

[2] S. Toyooka and Y. Iwaasa. Automatic profilometry of 3-D diffuse objects by spatial phase detection. *Applied Optics*, 25(10):1630–1633, 1986.

[3] Q. Hu, P. S. Huang, Q. Fu, and F. Chiang. Calibration of a three-dimensional shape measurement system. *Optical Engineering*, 42(2):487–493, 2003.

[4] P. S. Huang, Q. Hu, and F. Chiang. Error compensation for a three-dimensional shape measurement system. *Optical Engineering*, 42(2):482–486, 2003.

[5] R. Legarda-Sáenz, T. Bothe, and W. P. Jüptner. Accurate procedure for the calibration of a structured light system. *Optical Engineering*, 43(2):464–471, 2004.

[6] R. Sitnik, M. Kujawińska, and J. Woźnicki. Digital fringe projection system for large-volume 360-deg shape measurement. *Optical Engineering*, 41(2):443–449, 2002.

[7] K. Hibino, B. F. Oreb, D. I. Farrant, and K. G. Larkin. Phase shifting for nonsinusoidal waveforms with phase-shift errors. *Journal of the Optical Society of America A: Optics and Image Science, and Vision*, 12(4):761–768, 1995.

[8] Y. Surrel. Design of algorithms for phase measurements by the use of phase stepping. *Applied Optics*, 35(1):51–60, 1996.

[9] C. R. Coggrave and J. M. Huntley. Optimization of a shape measurement system based on spatial light modulators. *Optical Engineering*, 39(1):91–98, 2000.

[10] J. M. Huntley and H. O. Saldner. Temporal phase-unwrapping algorithm for automated interferogram analysis. *Applied Optics*, 32(17):3047–3052, 1993.

[11] H. Zhao, W. Chen, and Y. Tan. Phase-unwrapping algorithm for the measurement of three-dimensional object shapes. *Applied Optics*, 33(20):4497–4500, 1994.

[12] M. S. Mermelstein, D. L. Feldkhun, and L. G. Shirley. Video-rate surface profiling with acousto-optic accordion fringe interferometry. *Optical Engineering*, 39(1):106–113, 2000.

[13] D. C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[14] Camera calibration toolbox for Matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/.

[15] T. Peng and S. K. Gupta. Algorithms for generating adaptive projection patterns for 3-D shape measurement. In *Proceedings of IDETC/CIE 2007*, pages DETC2007–35452, 2007.